Revolution in education

Gone are the days of blackboards and textbooks if Richard Ennals' ideas catch on.

IF HISTORY is to be brought alive for students in the classroom they must be actively involved in school trips, research projects, class discussion or historical simulations. Some published simulations lack fine detail and adaptability, and do not allow students the scope to make decisions influencing the subsequent progress of the simulation.

The first attempts to develop a simulation capable of bridging the information gap used Basic for a simulation. It was tested in class, and was followed by a simulation of the Russian Revolution. Yet both models suffered from a number of crucial limitations.

The data description on individual characters was limited, not reflecting the richness of detail required. The process of interrogating and updating the database was relatively elaborate and preset, and only a finite range of queries and updatings was available. The database operated independently of the chronological program and the impact of decisions on the subsequent chronological program and state of the database was too crude and limited.

Prolog was then substituted for Basic and used to handle the same historical simulations working from identical source materials and accompanied by the same printed documentation. MicroProlog, a microcomputer implementation of Prolog for the Z-80 chip and the CP/M operating system, is available through Logic Programming Associates, 36 Gorst Road, London SW11 6JE. In these Prolog simulations the computer is used to facilitate decision and judgement on the part of the students thereby extending their capacity to consider information and see the effect of their decisions and actions on the network of which they form a part.

The briefing for individual characters in the Russian Revolution game was written before the development of computer-assisted versions of the simulation materials. The model for constructing such briefings is a network of relationships: a representation of the positions, wants, tactics and problems of 30 historical characters is presented using a common vocabulary for ease of developing and contrasting the information given. Participants should be able to function initially using the information given but be equipped with the language, constructs or notation with which to expand or change over the course of the simulation.

The form of the briefing information, issued on 30 separate briefing sheets, arises from the construction of complex semantic networks at the design stage used as a means of controlling the considerable volume of data. Translating such materials

into the form of a Prolog database is straightforward.

The Prolog database has immediate advantages for the participants. They are able to ask questions whose answers help them reach decisions. The questions are asked in Prolog and use the same formalism as the database. Participants can amend the database themselves by deleting or adding sentences.

The database is constructed in a simple notation whose interpretation in English is explicit. Economy of programming style is sacrificed in the cause of participation. For example, the program might contain the following information about the peasant:

```
Peasant wants bread
Peasant wants peace
Peasant wants land
English: Who wants better government?
Proglog: Which ((x) x wants better-
  government)
  Answer is (Soldier)
  Answer is (Sailor)
  Answer is (Peasant)
  Answer is (Factwker)
  No (more) answers
English: Who has tactics of non-
  cooperation?
Prolog: Which ((x) x tactics non-
  cooperation)
  Answer is (Lenin)
  Answer is (Tsar)
  Answer is (Tsarina)
  No (more) answers
English: Does Lenin support the Tsar?
Prolog: Does (Lenin supports Tsar)
No
```

To add new information regarding the peasant you have to add:

```
Add (Peasant supports Tsar)
```

The revised database can be saved at any stage of the simulation, provided other component programs are not loaded at the same time. Saving under a series of different names is recommended to avoid loss of information.

In a conventional simulation it is important to establish the historical context of the simulation in order to reduce the level of anachronism and misconception. Often an introduction is provided in the form of an opening lesson, a talk, film or printed materials. This will offer a framework upon which the simulation is to be constructed.

The Russian Revolution simulation is accompanied by printed introductory materials describing the state of Russia at the turn of the 20th century, Russian involvement in the First World War and contemporary social and political tensions. Altogether they provide a context for the events of 1917 which are to be reconstructed by the participants in the classroom.

Translating these three introductory pieces into Prolog programs is not difficult. A few names and details are discarded but the framework is faithfully recorded. Students can still have the original printed versions but the Prolog programs can directly affect the other components of the Russian Revolution package written in the same notation and with a common vocabulary. For example, historical background information is represented by

```
(Imperial-Russia mobilises army) date 1914
```

Then you can find out information about the different political parties:

(continued on next page)

(continued from previous page)

English: What did the Bolsheviks want?
Prolog: Which ((x) Bolsheviks want x)
Answer is (abolish-property)
Answer is (income-tax)
Answer is (nationalise-industry)
Answer is (free-education)
Answer is (world-revolution)
Answer is (remove-bourgeoisie)
Answer is (dictatorship-of-proletariat)
No (more) answers

No amendment of these programs is necessary. They are designed to apply constraints from the real world to the closed world of simulation, in a way that can visibly affect the logic of decision. In teaching history it is important to enable students to understand the reasons for the actions of historical agents. One way is to construct a practical inference, and Prolog simulation programs hold considerable promise for modelling and reconstructing events.

It is crucial to the success of any historical simulation that events and decisions can be located in a chronological context. To avoid needless anachronism it should be established what known historical events have preceded the situation under reconstruction. Similarly, during the simulation it is important to know what holds true at any given time as participants should be able to record their decisions and actions to form part of the framework for later decisions. It is remarkably easy to do this in Prolog.

Presented with the notation in Prolog students are able to record their own decisions and add their own information to the program. The notation conforms visually to English syntax though technically there are numerous differences. For instance, using the built-in relation Less, time-handling definitions can be built up as follows:

x lesseq x
x lesseq y if x LESS y
x during (y z) if y lesseq x and x lesseq z
x at y if x for (z X) and y during (z X)
(Bolsheviks oppose war) for (1 12)

If you are now at round 5 in this simulation and want to know what is known to hold true ask:

Which ((x) x at 5)
Answer is ((Bolsheviks oppose war))
No (more) answers

You might want to record a decision of a particpant at this stage:

Add ((Tsar orders tea) at 5)

It is then added to what is known to hold true so that if you ask the questions again there is a new answer:

Which ((x) x at 5)
Answer is ((Bolsheviks oppose war))
Answer is ((Tsar orders tea))
No (more) answers

The Prolog notation is easily adapted to take on the role of a simple expert system in directing such a simulation. At any given stage or round in the simulation the program can refer participants to primary or secondary historical sources. It can also emphasise particular issues, retrieve pieces of information, issue instructions or perhaps set problems to participants.

The program itself can be amended by the teacher to take account of the needs of his or her students. For instance, if you have reached round 7 you can type in:

Add (game round 7)

To find out what the key issue is thought to be, ask:

Which ((x) issue is x)
Answer is (reforms)
No (more) answers

To find out which of the printed documents to refer to at this stage ask:

Which ((x) see source x)
Answer is (24 25 26)
No (more) answers

As a refinement to this program you could add the idea that anything the Tsar orders at any time is an issue thereafter:

Add (issue is x if (Tsar orders x) at y and game round z and y lesseq z)

To see the effect of this addition ask:

Which ((x) issue is x)
Answer is (reforms)
Answer is (tea)
No (more) answers

MicroProlog will be ready early in 1983 for the Sinclair Spectrum and BBC Micro, and courses in using the language are being organised by the MEP. The book *Beginning MicroProlog* by the author of this article is published by Ellis Horwood Ltd and Heinemann Computers in Education Ltd at £6.50 in paperback and £12.50 in hardback. It includes recent developments of principles outlined in this article.

COMMENT - Roy Atherton

Richard Ennals' work is very significant. Prolog is a descriptive language. As can be seen from the article it is an excellent means of handling data in knowledge-based intelligent systems. Prolog is significant in three major ways for education.

(1) It is now available in a version especially simplified and field-tested with CSE classes. A syllabus has been developed.

(2) It is a link with the future in much the same way as old Basic is a link with the past. Fifth generation machines will be powerful, plentiful, portable, cheap, intelligent, knowledge-based systems. Prolog is a step towards that goal.

(3) Even the good process-oriented languages like Comal or other structured Basics are relatively weak on data handling of the kind most appropriate in school work across the curriculum. Prolog fills that gap and goes much further.

Time will determine the balance in schools between the use of good process-oriented languages (Comal or structured Basics) and descriptive languages (Prolog or Logo). Time will also determine whether essentially separate development continues or whether the present small overlap grows and causes a "coming together" of the two types.